A PRIMAL BARVINOK ALGORITHM BASED ON IRRATIONAL DECOMPOSITIONS

MATTHIAS KÖPPE

Abstract. We introduce variants of Barvinok's algorithm for counting lattice points in polyhedra. The new algorithms are based on irrational signed decomposition in the primal space and the construction of rational generating functions for cones with low index. We give computational results that show that the new algorithms are faster than the existing algorithms by a large factor.

1. Introduction

Twelve years have passed since Alexander Barvinok's amazing algorithm for counting lattice points in polyhedra was published (Barvinok, 1994). In the mean time, efficient implementations (De Loera et al., 2004b, Verdoolaege et al., 2005) were designed, which helped to make Barvinok's algorithm a practical tool in many applications in discrete mathematics. The implications of Barvinok's technique, of course, reach far beyond the domain of combinatorial counting problems: For example, De Loera et al. (2005b) pointed out applications in Integer Linear Programming, and De Loera et al. (2006a,b) obtained a fully polynomial-time approximation scheme (FPTAS) for optimizing arbitrary polynomial functions over the mixed-integer points in polytopes of fixed dimension.

Barvinok's algorithm first triangulates the supporting cones of all vertices of a polytope, to obtain simplicial cones. Then, the simplicial cones are recursively decomposed into unimodular cones. It is essential that one uses *signed decompositions* here; triangulating these cones is not good enough to give a polynomiality result. The rational generating functions of the resulting unimodular cones can then be written down easily. Adding and subtracting them according to the inclusion-exclusion principle and the theorem of Brion (1988) gives the rational generating function of the polytope. The number of lattice points in the polytope can finally be obtained by applying residue techniques on the rational generating function.

The algorithm in the original paper (Barvinok, 1994) worked explicitly with all the lower-dimensional cones that arise from the intersecting faces of the subcones in an inclusion-exclusion formula. Later it was pointed out that it is possible to simplify the algorithm by computing with full-dimensional

Date: Revision: 1.41 − Date: 2006/09/25 11:54:39 .

2000 *Mathematics Subject Classification.* 05A15; 52C07; 68W30.

This work was supported by a Feodor Lynen Research Fellowship from the Alexander von Humboldt Foundation. The author also acknowledges the hospitality of Jesús De Loera and the Department of Mathematics of the University of California, Davis. The author is grateful to David Einstein for pointing out a mistake in an earlier version of this paper.

cones only, by making use of Brion's "polarization trick" (see Barvinok and Pommersheim, 1999, Remark 4.3): The computations with rational generating functions are invariant with respect to the contribution of non-pointed cones (cones containing a non-trivial linear subspace). By operating in the dual space, i.e., by computing with the polars of all cones, lower-dimensional cones can be safely discarded, because this is equivalent to discarding non-pointed cones in the primal space. The practical implementations also rely heavily on this polarization trick.

In practical implementations of Barvinok's algorithm, one observes that in the hierarchy of cone decompositions, the index of the decomposed cones quickly descends from large numbers to fairly low numbers. The "last mile," i.e., decomposing many cones with fairly low index, creates a huge number of unimodular cones and thus is the bottleneck of the whole computation in many instances.

The idea of this paper is to stop the decomposition when the index of a cone is small enough, and to compute with generating functions for the integer points in cones of small index rather than unimodular cones. When we try to implement this simple idea in Barvinok's algorithm, as outlined in section 3, we face a major difficulty, however: Polarizing back a cone of small index can create a cone of very large index, because determinants of $d \times d$ matrices are homogeneous of order d.

To address this difficulty, we avoid polarization altogether and perform the signed decomposition in the primal space instead. To avoid having to deal with all the lower-dimensional subcones, we use the concept of *irrational decompositions* of rational polyhedra. Beck and Sottile (2005) introduced this notion to give astonishingly simple proofs for three theorems of Stanley on generating functions for the integer points in rational polyhedral cones. Using the same technique, Beck et al. (2005) gave simplified proofs of theorems of Brion and Lawrence–Varchenko. An irrational decomposition of a polyhedron is a decomposition into polyhedra whose proper faces do not contain any lattice points. Counting formulas for lattice points based on irrational decompositions therefore do not need to take any inclusion-exclusion principle into account.

We give an explicit construction of a *uniform irrational shifting vector* $\mathbf{s}$ for a cone $\mathbf{v} + K$ with apex $\mathbf{v}$ such that the shifted cone $(\mathbf{v} + \mathbf{s}) + K$ has the same lattice points and contains no lattice points on its proper faces (section 4). More strongly, we prove that *all cones* appearing in the signed decompositions of $(\mathbf{v} + \mathbf{s}) + K$ in Barvinok's algorithm contain no lattice points on their proper faces. Therefore, discarding lower-dimensional cones is safe. Despite its name, the vector $\mathbf{s}$ only has *rational coordinates*, so after shifting the cone by $\mathbf{s}$, large parts of existing implementations of Barvinok's algorithm can be reused to compute the irrational primal decompositions.

In section 5, we show the precise algorithm. We also show that the same technique can be applied to the "homogenized version" of Barvinok's algorithm that was proposed by De Loera et al. (2004a).

In section 6, we extend the irrationalization technique to non-simplicial cones. This gives rise to an "all-primal" Barvinok algorithm, where also triangulation of non-simplicial cones is performed in the primal space. This

allows us to handle problems where the triangulation of the dual cones is hard, e.g., in the case of cross polytopes.

Finally, in section 7, we report on computational results. Results on benchmark problems show that the new algorithms are faster than the existing algorithms by orders of magnitude. We also include results for problems that could not previously be solved with Barvinok techniques.

2. Barvinok's algorithm

Let $P \subseteq \mathbf{R}^d$ be a rational polyhedron. The *generating function* of $P \cap \mathbf{Z}^d$ is defined as the formal Laurent series

$$\tilde{g}_P(\mathbf{z}) = \sum_{\boldsymbol{\alpha} \in P \cap \mathbf{Z}^d} \mathbf{z}^{\boldsymbol{\alpha}} \in \mathbf{Z}[[z_1, \dots, z_d, z_1^{-1}, \dots, z_d^{-1}]],$$

using the multi-exponent notation $\mathbf{z}^{\boldsymbol{\alpha}} = \prod_{i=1}^d z_i^{\alpha_i}$. If P is bounded, $\tilde{g}_P$ is a Laurent polynomial, which we consider as a rational function g_P. If P is not bounded but is pointed (i.e., P does not contain a straight line), there is a non-empty open subset $U \subseteq \mathbf{C}^d$ such that the series converges absolutely and uniformly on every compact subset of U to a rational function g_P. If P contains a straight line, we set $g_P \equiv 0$. The rational function $g_P \in \mathbf{Q}(z_1, \dots, z_d)$ defined in this way is called the *rational generating function* of $P \cap \mathbf{Z}^d$.

Barvinok's algorithm computes the rational generating function of a polyhedron P. It proceeds as follows. By the theorem of Brion (1988), the rational generating function of a polyhedron can be expressed as the sum of the rational generating functions of the supporting cones of its vertices. Let $\mathbf{v}_i \in \mathbf{Q}^d$ be a vertex of the polyhedron P. Then the *supporting cone* $\mathbf{v}_i + C_i$ of $\mathbf{v}_i$ is the (shifted) polyhedral cone defined by $\mathbf{v}_i + \operatorname{cone}(P - \mathbf{v}_i)$. Every supporting cone $\mathbf{v}_i + C_i$ can be triangulated to obtain simplicial cones $\mathbf{v}_i + C_{ij}$. Let $K = \mathbf{v} + B\mathbf{R}^d_+$ be a simplicial full-dimensional cone, whose *basis vectors* $\mathbf{b}_1, \dots, \mathbf{b}_d$ (i.e., representatives of its extreme rays) are given by the columns of some matrix $B \in \mathbf{Z}^{d \times d}$. We assume that the basis vectors are primitive vectors of the standard lattice $\mathbf{Z}^d$. Then the *index* of K is defined to be $\operatorname{ind} K = |\det B|$; it can also be interpreted as the cardinality of $\Pi \cap \mathbf{Z}^d$, where Π is the *fundamental parallelepiped* of K, i.e., the half-open parallelepiped

$$\Pi = \mathbf{v} + \left\{ \textstyle\sum_{i=1}^d \lambda_i \mathbf{b}_i : 0 \le \lambda_i < 1 \right\}.$$

We remark that the set $\Pi \cap \mathbf{Z}^d$ can also be seen as a set of representatives of the cosets of the lattice $B\mathbf{Z}^d$ in the standard lattice $\mathbf{Z}^d$; we shall make use of this interpretation in section 3. Barvinok's algorithm now computes a *signed decomposition* of the simplicial cone K to produce other simplicial cones with smaller index. To this end, the algorithm constructs a vector $\mathbf{w} \in \mathbf{Z}^d$ such that

$$\mathbf{w} = \alpha_1 \mathbf{b}_1 + \dots + \alpha_d \mathbf{b}_d \quad \text{with} \quad |\alpha_i| \le |\det B|^{-1/d} \le 1. \tag{1}$$

This can be accomplished using integer programming or lattice basis reduction. The cone is then decomposed into cones spanned by d vectors from the set $\{\mathbf{b}_1, \dots, \mathbf{b}_d, \mathbf{w}\}$; each of the resulting cones then has an index bounded above by $(\operatorname{ind} K)^{(d-1)/d}$. In general, these cones form a signed decomposition

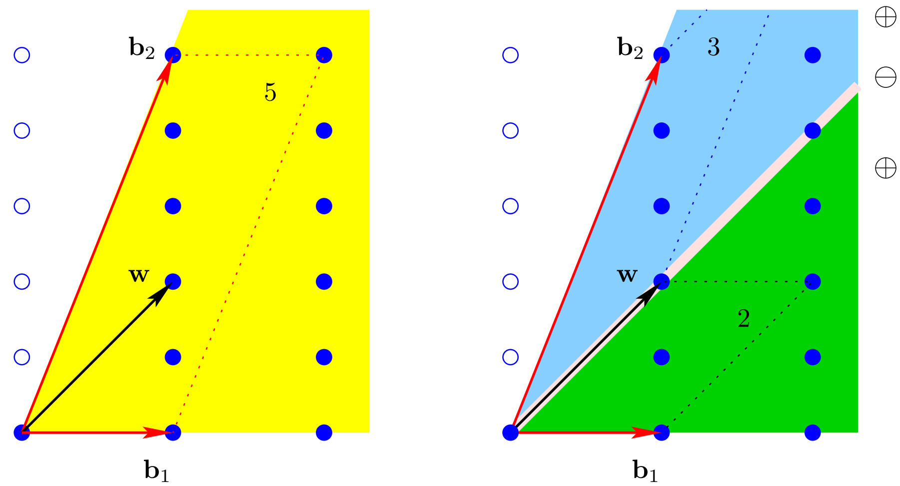

Figure 1. A triangulation of the cone of index 5 generated by $\mathbf{b}^1$ and $\mathbf{b}^2$ into the two cones spanned by $\{\mathbf{b}^1, \mathbf{w}\}$ and $\{\mathbf{b}^2, \mathbf{w}\}$, having an index of 2 and 3, respectively. We have the inclusion-exclusion formula $g_{\mathrm{cone}\{\mathbf{b}_1,\mathbf{b}_2\}}(\mathbf{z}) = g_{\mathrm{cone}\{\mathbf{b}_1,\mathbf{w}\}}(\mathbf{z}) + g_{\mathrm{cone}\{\mathbf{b}_2,\mathbf{w}\}}(\mathbf{z}) - g_{\mathrm{cone}\{\mathbf{w}\}}(\mathbf{z})$; here the one-dimensional cone spanned by $\mathbf{w}$ needed to be subtracted.

of K (see Figure 2); if $\mathbf{w}$ lies inside K, they form a triangulation of K (see Figure 1). The resulting cones and their intersecting proper faces (arising in an inclusion-exclusion formula) are recursively processed, until *unimodular* cones, i.e., cones of index 1 are obtained. Finally, for a unimodular cone $\mathbf{v} + B\mathbf{R}^d_+$, the rational generating function can be easily written down as

$$\frac{\mathbf{z}^{\mathbf{a}}}{\prod_{j=1}^{d}(1 - \mathbf{z}^{\mathbf{b}_j})}, \tag{2}$$

where $\mathbf{a}$ is the unique integer point in the fundamental parallelepiped of the cone. We summarize Barvinok's algorithm below.

Algorithm 1 (Barvinok's original (primal) algorithm)**.**
Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities.
Output: The rational generating function for $P \cap \mathbf{Z}^d$ in the form

$$g_P(\mathbf{z}) = \sum_{i\in I} \epsilon_i \frac{\mathbf{z}^{\mathbf{a}_i}}{\prod_{j=1}^{d}(1 - \mathbf{z}^{\mathbf{b}_{ij}})} \tag{3}$$

where $\epsilon_i \in \{\pm 1\}$, $\mathbf{a}_i \in \mathbf{Z}^d$, and $\mathbf{b}_{ij} \in \mathbf{Z}^d$.

1. Compute all vertices $\mathbf{v}_i$ and corresponding supporting cones C_i of P.
2. Triangulate C_i into simplicial cones C_{ij}, keeping track of all the intersecting proper faces.
3. Apply signed decomposition to the cones $\mathbf{v}_i + C_{ij}$ to obtain unimodular cones $\mathbf{v}_i + C_{ijl}$, keeping track of all the intersecting proper faces.
4. Compute the unique integer point $\mathbf{a}_i$ in the fundamental parallelepiped of every resulting cone $\mathbf{v}_i + C_{ijl}$.
5. Write down the formula (3).

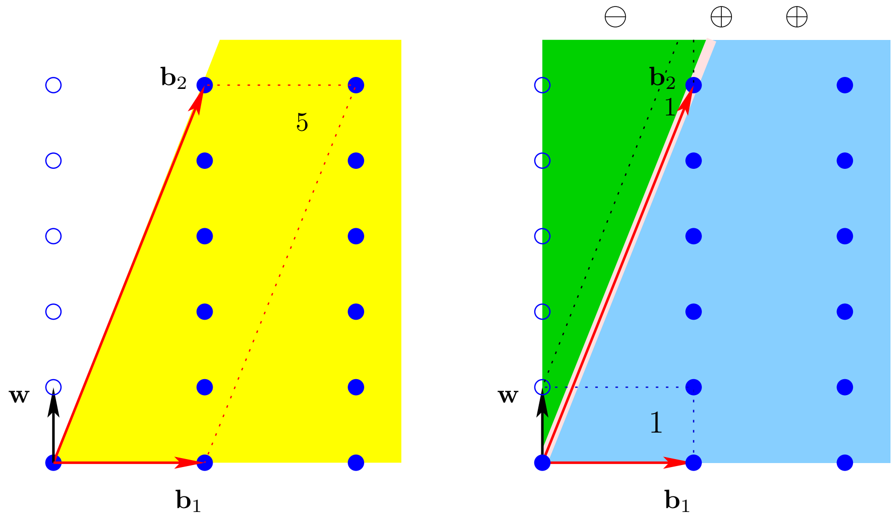

FIGURE 2. A signed decomposition of the cone of index 5 generated by $\mathbf{b}^1$ and $\mathbf{b}^2$ into the two unimodular cones spanned by $\{\mathbf{b}^1, \mathbf{w}\}$ and $\{\mathbf{b}^2, \mathbf{w}\}$. We have the inclusion-exclusion formula $g_{\operatorname{cone}\{\mathbf{b}_1,\mathbf{b}_2\}}(\mathbf{z}) = g_{\operatorname{cone}\{\mathbf{b}_1,\mathbf{w}\}}(\mathbf{z}) - g_{\operatorname{cone}\{\mathbf{b}_2,\mathbf{w}\}}(\mathbf{z}) + g_{\operatorname{cone}\{\mathbf{w}\}}(\mathbf{z})$.

The recursive decomposition of cones defines a *decomposition tree*. Due to the descent of the indices in the signed decomposition procedure, the following estimate holds for its depth:

Lemma 2 (Barvinok, 1994)**.** *Let $B\mathbf{R}^d_+$ be a simplicial full-dimensional cone, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let $D = |\det B|$. Then the depth of the decomposition tree is at most*

$$k(D) = \left\lfloor 1 + \frac{\log_2 \log_2 D}{\log_2 \frac{d}{d-1}} \right\rfloor. \tag{4}$$

Because at each decomposition step at most $\mathrm{O}(2^d)$ cones are created and the depth of the tree is doubly logarithmic in the index of the input cone, Barvinok could obtain a polynomiality result *in fixed dimension*:

Theorem 3 (Barvinok, 1994)**.** *Let d be fixed. There exists a polynomial-time algorithm for computing the rational generating function of a polyhedron $P \subseteq \mathbf{R}^d$ given by rational inequalities.*

Later the algorithm was improved by making use of Brion's "polarization trick" (see Barvinok and Pommersheim, 1999, Remark 4.3): The computations with rational generating functions are invariant with respect to the contribution of non-pointed cones (cones containing a non-trivial linear subspace). The reason is that the rational generating function of every non-pointed cone is zero. By operating in the dual space, i.e., by computing with the polars of all cones, lower-dimensional cones can be safely discarded, because this is equivalent to discarding non-pointed cones in the primal space.

Algorithm 4 (Dual Barvinok algorithm)**.**
Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities.
Output: The rational generating function for $P \cap \mathbf{Z}^d$ in the form

$$g_P(\mathbf{z}) = \sum_{i\in I} \epsilon_i \frac{\mathbf{z}^{\mathbf{a}_i}}{\prod_{j=1}^{d}(1-\mathbf{z}^{\mathbf{b}_{ij}})} \tag{5}$$

where $\epsilon_i \in \{\pm 1\}$, $\mathbf{a}_i \in \mathbf{Z}^d$, and $\mathbf{b}_{ij} \in \mathbf{Z}^d$.

1. Compute all vertices $\mathbf{v}_i$ and corresponding supporting cones C_i of P.
2. Polarize the supporting cones C_i to obtain C_i°.
3. Triangulate C_i° into simplicial cones C_{ij}°, discarding lower-dimensional cones.
4. Apply Barvinok's signed decomposition to the cones $\mathbf{v}_i + C_{ij}^\circ$ to obtain cones $\mathbf{v}_i + C_{ijl}^\circ$, stopping decomposition when a unimodular cone is obtained. Discard all lower-dimensional cones.
5. Polarize back C_{ijl}° to obtain cones C_{ijl}.
6. Compute the unique integer point $\mathbf{a}_i$ in the fundamental parallelepiped of every resulting cone $\mathbf{v}_i + C_{ijl}$.
7. Write down the formula (5).

This variant of the algorithm is much faster than the original algorithm because in each step of the signed decomposition at most d, rather than $\mathrm{O}(2^d)$, cones are created. The practical implementations LattE (De Loera et al., 2004b) and `barvinok` (Verdoolaege et al., 2005) also rely heavily on this polarization trick.

3. The Barvinok algorithm with stopped decomposition

We start out by introducing a first variant of Barvinok's algorithm that stops decomposing cones before unimodular cones are reached. As we will see in the computational results in section 7, already the simple modification that we propose can give a significant improvement of the running time for some problems, at least in low dimension.

Algorithm 5 (Dual Barvinok algorithm with stopped decomposition)**.**
Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities; the maximum index ℓ.
Output: The rational generating function for $P \cap \mathbf{Z}^d$ in the form

$$g_P(\mathbf{z}) = \sum_{i\in I} \epsilon_i \frac{\sum_{\mathbf{a}\in A_i} \mathbf{z}^{\mathbf{a}}}{\prod_{j=1}^{d}(1-\mathbf{z}^{\mathbf{b}_{ij}})} \tag{6}$$

where $\epsilon_i \in \{\pm 1\}$, $A_i \subseteq \mathbf{Z}^d$ with $|A_i| \le \ell$, and $\mathbf{b}_{ij} \in \mathbf{Z}^d$.

1. Compute all vertices $\mathbf{v}_i$ and corresponding supporting cones C_i of P.
2. Polarize the supporting cones C_i to obtain C_i°.
3. Triangulate C_i° into simplicial cones C_{ij}°, discarding lower-dimensional cones.
4. Apply Barvinok's signed decomposition to the cones $\mathbf{v}_i + C_{ij}^\circ$ to obtain cones $\mathbf{v}_i + C_{ijl}^\circ$, stopping decomposition when a polarized-back cone $C_{ijl} = (C_{ijl}^\circ)^\circ$ has index at most ℓ. Discard all lower-dimensional cones.
5. Polarize back C_{ijl}° to obtain cones C_{ijl}.

6. Enumerate the integer points in the fundamental parallelepipeds of all resulting cones $\mathbf{v}_i + C_{ijl}$ to obtain the sets A_i.
7. Write down the formula (6).

As mentioned above, the integer points in the fundamental parallelepiped of a cone $\mathbf{v}_i + B_{ijl}\mathbf{R}^d_+$ can be interpreted as representatives of the cosets of the lattice $B_{ijl}\mathbf{Z}^d$ in the standard lattice $\mathbf{Z}^d$. Hence they can be easily enumerated in step 6 by computing the Smith normal form of the generator matrix B_{ijl}; see Lemma 5.2 of Barvinok (1993). The Smith normal form can be computed in polynomial time, even if the dimension is not fixed (Kannan and Bachem, 1979).

We remark that both triangulation and signed decomposition are done in the dual space, but the stopping criterion is the index of the polarized-back cones (in the primal space). The reason for this stopping criterion is that we wish to control the maximum number of points in the fundamental parallelepipeds that need to be enumerated. Indeed, when the maximum ℓ is chosen as a constant or polynomially in the input size, then Algorithm 5 clearly runs in polynomial time (in fixed dimension).

Each step of Barvinok's signed decomposition reduces the index of the decomposed cones. When the index of a cone C°_{ijl} is Δ, in the worst case the polarized-back cone C_{ijl} has index Δ^{d-1}, where d is the dimension. If the dimension is too large, the algorithm often needs to decompose cones down to a very low index or even index 1, so the speed-up of the algorithm will be very limited. This can be seen from the computational results in section 7.

4. Construction of a uniform irrational shifting vector

In this section, we will give an explicit construction of an *irrational shifting vector* $\mathbf{s}$ for a simplicial cone $\mathbf{v} + K$ with apex $\mathbf{v}$ such that the shifted cone $(\mathbf{v} + \mathbf{s}) + K$ has the same lattice points and contains no lattice points on its proper faces. The "irrationalization" (or perturbation) will be *uniform* in the sense that also every cone arising during the Barvinok decomposition does not contain any lattice points on its proper faces. This will enable us to perform the Barvinok decomposition in the primal space, discarding all lower-dimensional cones.

To accomplish this goal, we shall first describe a subset of the *stability region* of a cone $\mathbf{v} + K$ with apex at $\mathbf{v}$, i.e., the set of apex points $\tilde{\mathbf{v}}$ such that $\tilde{\mathbf{v}} + K$ contains the same lattice points as $\mathbf{v} + K$; see Figure 3.

Lemma 6 (Stability cube). *Let $\mathbf{v} + B\mathbf{R}^d_+$ be a simplicial full-dimensional cone with apex at $\mathbf{v} \in \mathbf{Q}^d$, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let $\mathbf{b}^*_1, \dots, \mathbf{b}^*_d$ be a basis of the dual cone, given by the columns of the matrix $B^* = -(B^{-1})^\top$.*

Let $D = |\det B|$. Let $\boldsymbol{\lambda} \in \mathbf{Q}^d$ and $\hat{\boldsymbol{\lambda}} \in \mathbf{Q}^d$ be defined by

$$\lambda_i = \langle \mathbf{b}^*_i, \mathbf{v}\rangle \quad \text{and} \quad \hat{\lambda}_i = \frac{1}{D}\left(\lfloor D\lambda_i \rfloor + \frac{1}{2}\right) \quad \text{for } i = 1, \dots, d.$$

Let

$$\hat{\mathbf{v}} = -B\hat{\boldsymbol{\lambda}} \quad \text{and} \quad \rho = \frac{1}{2D \cdot \max_{i=1}^d \|\mathbf{b}^*_i\|_1}.$$

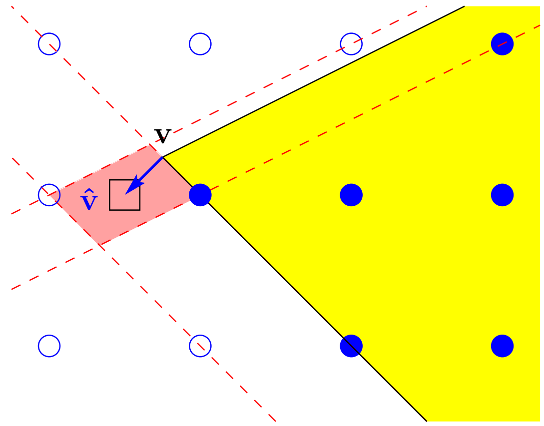

Figure 3. The stability region of a cone

Then, for every $\tilde{\mathbf{v}}$ *with* $\|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty < \rho$*, the cone* $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$ *contains the same integer points as the cone* $\mathbf{v} + B\mathbf{R}^d_+$ *and does not have integer points on its proper faces.*

In the proof of the lemma, we will use of the inequality description (H-representation) of the simplicial cone $\mathbf{v} + B\mathbf{R}^d_+$. It is given by the basis vectors of the dual cone:

$$\mathbf{v} + B\mathbf{R}^d_+ = \Big\{ \mathbf{x} \in \mathbf{R}^d : \langle \mathbf{b}^*_i, \mathbf{x} \rangle \leq \langle \mathbf{b}^*_i, \mathbf{v} \rangle \text{ for } i = 1, \dots, d \Big\}. \tag{7}$$

Proof of Lemma 6. Let $\tilde{\boldsymbol{\lambda}}$ be defined by $\tilde{\lambda}_i = \langle \mathbf{b}^*_i, \tilde{\mathbf{v}} \rangle$. Then we have

$$\big|\tilde{\lambda}_i - \hat{\lambda}_i\big| \leq \|\mathbf{b}^*_i\|_1 \cdot \|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty < \|\mathbf{b}^*_i\|_1 \cdot \rho \leq \frac{1}{2D}. \tag{8}$$

By (7), a point $\mathbf{x} \in \mathbf{Z}^d$ lies in the cone $\mathbf{v} + B\mathbf{R}^d_+$ if and only if

$$\langle \mathbf{b}^*_i, \mathbf{x} \rangle \leq \langle \mathbf{b}^*_i, \mathbf{v} \rangle = \lambda_i \quad \text{for } i = 1, \dots, d.$$

Likewise, $\mathbf{x} \in \tilde{\mathbf{v}} + B\mathbf{R}^d_+$ if and only if

$$\langle \mathbf{b}^*_i, \mathbf{x} \rangle \leq \langle \mathbf{b}^*_i, \tilde{\mathbf{v}} \rangle = \tilde{\lambda}_i \quad \text{for } i = 1, \dots, d.$$

Note that for $\mathbf{x} \in \mathbf{Z}^d$, the left-hand sides of both inequalities are an integer multiple of $\frac{1}{D}$. Therefore, we obtain equivalent statements by rounding down the right-hand sides to integer multiples of $\frac{1}{D}$. For the right-hand side of (4) we have by (8)

$$\tilde{\lambda}_i = \hat{\lambda}_i + (\tilde{\lambda}_i - \hat{\lambda}_i) < \frac{1}{D}\left(\lfloor D\lambda_i \rfloor + \frac{1}{2}\right) + \frac{1}{2D} = \frac{1}{D}\lfloor D\lambda_i \rfloor + 1, \tag{9a}$$

$$\tilde{\lambda}_i = \hat{\lambda}_i + (\tilde{\lambda}_i - \hat{\lambda}_i) > \frac{1}{D}\left(\lfloor D\lambda_i \rfloor + \frac{1}{2}\right) - \frac{1}{2D} = \frac{1}{D}\lfloor D\lambda_i \rfloor, \tag{9b}$$

so λ_i and $\tilde{\lambda}_i$ are rounded down to the same value $\frac{1}{D}\lfloor D\lambda_i \rfloor$. Thus, the cone $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$ contains the same integer points as the cone $\mathbf{v} + B\mathbf{R}^d_+$. Moreover, since the inequalities (9) are strict, the cone $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$ does not have integer points on its proper faces. ∎

For non-simplicial cones, we will give an algorithmic construction for a stability cube in section 6.

Next we make use of the estimate for the depth of the decomposition tree in Barvinok's algorithm given in Lemma 2. On each level of the decomposition, the entries in the basis matrices can grow, but not by much. We obtain:

Lemma 7. *Let $B\mathbf{R}^d_+$ be a simplicial full-dimensional cone, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let $D = |\det B|$. Let $C \in \mathbf{Z}_+$ be a number such that $|B_{i,j}| \leq C$.*

Then all the basis matrices $\bar{B}$ of the cones that appear in the recursive signed decomposition procedure of Barvinok's algorithm applied to $B\mathbf{R}^d_+$ have entries bounded above by $d^{k(D)}C$, where $k(D)$ is defined by (4).

Proof. Given a cone spanned by the columns $\mathbf{b}_1, \dots, \mathbf{b}_d$ of the matrix B, Barvinok's algorithm constructs a vector $\mathbf{w} \in \mathbf{Z}^d$ such that

$$\mathbf{w} = \alpha_1 \mathbf{b}_1 + \cdots + \alpha_d \mathbf{b}_d \quad \text{with} \quad |\alpha_i| \leq |\det B|^{-1/d} \leq 1. \tag{10}$$

Thus $\|\mathbf{w}\|_\infty \leq dC$. The cone is then decomposed into cones spanned by d vectors from the set $\{\mathbf{b}_1, \dots, \mathbf{b}_d, \mathbf{w}\}$. Thus the entries in the corresponding basis matrices are bounded by dC. The result follows then by Lemma 2. □

If we can bound the entries of an integer matrix with non-zero determinant, we can also bound the entries of its inverse.

Lemma 8. *Let $B \in \mathbf{Z}^{d\times d}$ be a matrix with $|B_{i,j}| \leq C$. Let $D = |\det B|$. Then the absolute values of the entries of B^{-1} are bounded above by*

$$\frac{1}{D}(d-1)!C^{d-1}.$$

Proof. We have $\left|(B^{-1})_{k,l}\right| = \frac{1}{D}\left|\det B_{(k,l)}\right|$, where $B_{(k,l)}$ is the matrix obtained from deleting the k-th row and l-th column from B. Now the desired estimate follows from a formula for $\det B_{(k,l)}$ and from $|B_{i,j}| \leq C$. □

Thus, we obtain a bound on the norm of the basis vectors of the polars of all cones occurring in the signed decomposition procedure of Barvinok's algorithm.

Corollary 9 (A bound on the dual basis vectors)**.** *Let $B\mathbf{R}^d_+$ be a simplicial full-dimensional cone, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let $D = |\det B|$. Let C be a number such that $|B_{i,j}| \leq C$.*

Let $\bar{B}^* = -(\bar{B}^{-1})^\top$ be the basis matrix of the polar of an arbitrary cone $\bar{B}\mathbf{R}^d_+$ that appears in the recursive signed decomposition procedure applied to $B\mathbf{R}^d_+$. Then, for every column vector $\bar{\mathbf{b}}^*_i$ of $\bar{B}^*$ we have the estimate

$$\left\|\det \bar{B} \cdot \mathbf{b}^*_i\right\|_\infty \leq (d-1)!\left(d^{k(D)}C\right)^{d-1} =: L \tag{11}$$

where $k(D)$ is defined by (4).

Proof. By Lemma 7, the entries of $\bar{B}$ are bounded above by $d^{k(D)}C$. Then the result follows from Lemma 8. □

The construction of the "irrational" shifting vector is based on the following lemma.

Lemma 10 (The irrational lemma)**.** *Let $M \in \mathbf{Z}_+$ be an integer. Let*

$$\mathbf{q} = \left(\frac{1}{2M}, \frac{1}{(2M)^2}, \dots, \frac{1}{(2M)^d} \right). \tag{12}$$

Then $\langle \mathbf{c}, \mathbf{q} \rangle \notin \mathbf{Z}$ for every $\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}$ with $\|\mathbf{c}\|_\infty < M$.

Proof. Follows from the principle of representations of rational numbers in a positional system of base $2M$. $\square$

Theorem 11. *Let $\mathbf{v} + B\mathbf{R}^d_+$ be a simplicial full-dimensional cone with apex at $\mathbf{v} \in \mathbf{Q}^d$, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let $D = |\det B|$, C be a number such that $|B_{i,j}| \le C$ and let $\hat{\mathbf{v}} \in \mathbf{Q}^d$ and $\rho \in \mathbf{Q}_+$ be the data from Lemma 6 describing the stability cube of $\mathbf{v} + B\mathbf{R}^d_+$. Let $0 < r \in \mathbf{Z}$ such that $r^{-1} < \rho$. Using*

$$k = \left\lfloor 1 + \frac{\log_2 \log_2 D}{\log_2 \frac{d}{d-1}} \right\rfloor, \tag{13}$$

$L = (d-1)!(d^k C)^{d-1}$, and $M = 2L$, define

$$\mathbf{s} = \frac{1}{r} \cdot \left(\frac{1}{(2M)^1}, \frac{1}{(2M)^2}, \dots, \frac{1}{(2M)^d} \right).$$

Finally let $\tilde{\mathbf{v}} = \hat{\mathbf{v}} + \mathbf{s}$.

(i) *We have $(\tilde{\mathbf{v}} + B\mathbf{R}^d_+) \cap \mathbf{Z}^d = (\mathbf{v} + B\mathbf{R}^d_+) \cap \mathbf{Z}^d$, i.e., the shifted cone has the same set of integer points as the original cone.*
(ii) *The shifted cone $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$ contains no lattice points on its proper faces.*
(iii) *More strongly,* all cones *appearing in the signed decompositions of the shifted cone $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$ in Barvinok's algorithm contain no lattice points on their proper faces.*

Proof. Part (i). This follows from Lemma 6 because $\tilde{\mathbf{v}}$ clearly lies in the open stability cube.

Parts (ii) and (iii). Every cone appearing in the course of Barvinok's signed decomposition algorithm has the same apex $\tilde{\mathbf{v}}$ as the input cone and a basis $\bar{B} \in \mathbf{Z}^{d\times d}$ with $\left|\det \bar{B}\right| \le D$. Let such a $\bar{B}$ be fixed and denote by $\bar{\mathbf{b}}_i^*$ the columns of the dual basis matrix $\bar{B}^* = -(\bar{B}^{-1})^\top$. Let $\mathbf{z} \in \mathbf{Z}^d$ be an arbitrary integer point. We shall show that $\mathbf{z}$ is not on any of the facets of the cone, i.e.,

$$\langle \bar{\mathbf{b}}_i^*, \mathbf{z} - \tilde{\mathbf{v}} \rangle \neq 0 \qquad \text{for } i = 1, \dots, d. \tag{14}$$

Let $i \in \{1, \dots, d\}$ arbitrary. We will show (14) by proving that

$$\langle \det \bar{B} \cdot \bar{\mathbf{b}}_i^*, \tilde{\mathbf{v}} \rangle \notin \mathbf{Z}. \tag{15}$$

Clearly, if (15) holds, we have $\langle \bar{\mathbf{b}}_i^*, \tilde{\mathbf{v}} \rangle \notin (\det \bar{B})^{-1}\mathbf{Z}$. But since $\langle \bar{\mathbf{b}}_i^*, \mathbf{z} \rangle \in (\det \bar{B})^{-1}\mathbf{Z}$, we have $\langle \bar{\mathbf{b}}_i^*, \mathbf{z} - \tilde{\mathbf{v}} \rangle \notin \mathbf{Z}$; in particular it is nonzero, which proves (14).

To prove (15), let $\mathbf{c} = \det \bar{B} \cdot \bar{\mathbf{b}}_i^*$. By Corollary 9, we have $\|\mathbf{c}\|_\infty \le L < M$. Now Lemma 10 gives $\langle \mathbf{c}, \mathbf{s} \rangle \notin \frac{1}{r}\mathbf{Z}$. Observing that by the definition (6), we have

$$\langle \mathbf{c}, \hat{\mathbf{v}} \rangle = \langle \mathbf{c}, -B\hat{\boldsymbol{\lambda}} \rangle \in \tfrac{1}{r}\mathbf{Z}.$$

Therefore, we have $\langle \mathbf{c}, \tilde{\mathbf{v}} \rangle = \langle \mathbf{c}, \hat{\mathbf{v}} + \mathbf{s} \rangle \notin \frac{1}{r}\mathbf{Z}$. This proves (15), and thus completes the proof. $\square$

5. The irrational algorithms

The following is our variant of the Barvinok algorithm.

Algorithm 12 (Primal irrational Barvinok algorithm)**.**

Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities; the maximum index ℓ.

Output: The rational generating function for $P \cap \mathbf{Z}^d$ in the form

$$g_P(\mathbf{z}) = \sum_{i \in I} \epsilon_i \frac{\sum_{\mathbf{a} \in A_i} \mathbf{z}^{\mathbf{a}}}{\prod_{j=1}^{d} (1 - \mathbf{z}^{\mathbf{b}_{ij}})} \tag{16}$$

where $\epsilon_i \in \{\pm 1\}$, $A_i \subseteq \mathbf{Z}^d$ with $|A_i| \leq \ell$, and $\mathbf{b}_{ij} \in \mathbf{Z}^d$.

1. Compute all vertices $\mathbf{v}_i$ and corresponding supporting cones C_i of P.
2. Polarize the supporting cones C_i to obtain C_i°.
3. Triangulate C_i° into simplicial cones C_{ij}°, discarding lower-dimensional cones.
4. Polarize back C_{ij}° to obtain simplicial cones C_{ij}.
5. Irrationalize all cones by computing new apex vectors $\tilde{\mathbf{v}}_{ij} \in \mathbf{Q}^d$ from $\mathbf{v}_i$ and C_{ij} as in Theorem 11.
6. Apply Barvinok's signed decomposition to the cones $\tilde{\mathbf{v}}_{ij} + C_{ij}$, discarding lower-dimensional cones, until all cones have index at most ℓ.
7. Enumerate the integer points in the fundamental parallelepipeds of all resulting cones to obtain the sets A_i.
8. Write down the formula (16).

Theorem 13. *Algorithm 12 is correct and runs in polynomial time when the dimension d is fixed and the maximum index ℓ is bounded by a polynomial in the input size.*

Proof. This is an immediate consequence of the analysis of Barvinok's algorithm. The irrationalization (step 5 of the algorithm) increases the encoding length of the apex vector only by a polynomial amount, because the dimension d is fixed and the depth k only depends doubly logarithmic on the initial index of the cone. $\square$

The same technique can also be applied to the "homogenized version" of Barvinok's algorithm that was proposed by De Loera et al. (2004a); see also De Loera et al. (2004b, Algorithm 11).

Algorithm 14 (Irrational homogenized Barvinok algorithm)**.**

Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities in the form $A\mathbf{x} \leq \mathbf{b}$; the maximum index ℓ.

Output: A rational generating function in the form (16) for the integer points in the *homogenization* of P, i.e., the cone

$$C = \{\, (\xi \mathbf{x}, \xi) : \mathbf{x} \in P,\, \xi \in \mathbf{R}_+ \,\}. \tag{17}$$

1. Consider the inequality description for C; it is given by $A\mathbf{x} - \mathbf{b}\xi \leq 0$. The polar C° then has the rays $(A_{i,\cdot}, -b_i)$, $i = 1, \dots, m$.
2. Triangulate C° into simplicial cones C_j°, discarding lower-dimensional cones.
3. Polarize back the cones C_j° to obtain simplicial cones C_j.

4. Irrationalize the cones C_j to obtain shifted cones $\tilde{\mathbf{v}}_j + C_j$.
5. Apply Barvinok's signed decomposition to the cones $\tilde{\mathbf{v}}_j + C_j$, discarding lower-dimensional cones, until all cones have index at most ℓ.
6. Write down the generating function.

6. Extension to the non-simplicial case

For polyhedral cones with few rays and many facets, it is usually much faster to perform triangulation in the primal space than in the dual space, cf. Büeler et al. (2000). In this section, we show how to perform both Barvinok decomposition and triangulation in the primal space.

The key idea is to use linear programming to compute a subset of the stability region of the non-simplicial cones.

Lemma 15. *There is a polynomial-time algorithm that, given the vertex $\mathbf{v} \in \mathbf{Q}^d$ and the facet vectors $\mathbf{b}_i^* \in \mathbf{Z}^d$, $i = 1, \dots, m$, of a full-dimensional polyhedral cone $C = \mathbf{v} + B\mathbf{R}^n_+$, where $n \geq d$, computes a point $\hat{\mathbf{v}} \in \mathbf{Q}^d$ and a positive scalar $\rho \in \mathbf{Q}$ such that for every $\tilde{\mathbf{v}}$ in the open cube with $\|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty < \rho$, the cone $\tilde{\mathbf{v}} + B\mathbf{R}^n_+$ has no integer points on its proper faces and contains the same integer points as $\mathbf{v} + B\mathbf{R}^n_+$.*

Proof. We maximize ρ subject to the linear inequalities

$$\langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \|\mathbf{b}_i^*\|_1 \,\rho \leq \lfloor \langle \mathbf{b}_i^*, \mathbf{v} \rangle \rfloor + 1, \tag{18a}$$

$$-\langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \|\mathbf{b}_i^*\|_1 \,\rho \leq - \lfloor \langle \mathbf{b}_i^*, \mathbf{v} \rangle \rfloor , \tag{18b}$$

where $\hat{\mathbf{v}} \in \mathbf{R}^d$ and $\rho \in \mathbf{R}_+$. We can solve this linear optimization problem in polynomial time. Let $(\hat{\mathbf{v}}, \rho)$ be an optimal solution. Let $\tilde{\mathbf{v}} \in \mathbf{R}^d$ with $\|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty < \rho$. Let $\mathbf{x} \in (\tilde{\mathbf{v}} + B\mathbf{R}^d_+) \cap \mathbf{Z}^d$. Then we have for every $i \in \{1, \dots, m\}$

$$\begin{aligned}
\langle \mathbf{b}_i^*, \mathbf{x} \rangle &\leq \langle \mathbf{b}_i^*, \tilde{\mathbf{v}} \rangle \\
&= \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \langle \mathbf{b}_i^*, \tilde{\mathbf{v}} - \hat{\mathbf{v}} \rangle \\
&\leq \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \|\mathbf{b}_i^*\|_1 \, \|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty \\
&< \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \|\mathbf{b}_i^*\|_1 \,\rho \\
&\leq \lfloor \langle \mathbf{b}_i^*, \mathbf{v} \rangle \rfloor + 1 && \text{by (18a).}
\end{aligned}$$

Because $\langle \mathbf{b}_i^*, \mathbf{x} \rangle$ is integer, we actually have $\langle \mathbf{b}_i^*, \mathbf{x} \rangle \leq \lfloor \langle \mathbf{b}_i^*, \mathbf{v} \rangle \rfloor$. Thus, $\mathbf{x}$ lies in the cone $\mathbf{v} + B\mathbf{R}^d_+$. Conversely, let $\mathbf{x} \in (\mathbf{v} + B\mathbf{R}^d_+) \cap \mathbf{Z}^d$. Then, for every $i \in \{1, \dots, m\}$, we have $\langle \mathbf{b}_i^*, \mathbf{x} \rangle \leq \langle \mathbf{b}_i^*, \mathbf{v} \rangle$. Since $\mathbf{x} \in \mathbf{Z}^d$, we can round down the right-hand side and obtain

$$\begin{aligned}
\langle \mathbf{b}_i^*, \mathbf{x} \rangle &\leq \lfloor \langle \mathbf{b}_i^*, \mathbf{v} \rangle \rfloor \\
&\leq \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle - \|\mathbf{b}_i^*\|_1 \,\rho && \text{by (18b)} \\
&< \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle - \|\mathbf{b}_i^*\|_1 \, \|\tilde{\mathbf{v}} - \hat{\mathbf{v}}\|_\infty \\
&\leq \langle \mathbf{b}_i^*, \hat{\mathbf{v}} \rangle + \langle \mathbf{b}_i^*, \tilde{\mathbf{v}} - \hat{\mathbf{v}} \rangle \\
&= \langle \mathbf{b}_i^*, \tilde{\mathbf{v}} \rangle.
\end{aligned}$$

Thus, $\mathbf{x} \in \tilde{\mathbf{v}} + B\mathbf{R}^d_+$. Moreover, since the inequality is strict, $\mathbf{x}$ does not lie on the face $\langle \mathbf{b}_i^*, \mathbf{x} \rangle = \langle \mathbf{b}_i^*, \tilde{\mathbf{v}} \rangle$ of the cone $\tilde{\mathbf{v}} + B\mathbf{R}^d_+$. □

Lemma 16 (Bound on the index of all subcones)**.** *Let $\mathbf{b}_i \in \mathbf{Z}^d$, $i = 1, \dots, n$, be the generators of a full-dimensional polyhedral cone $K \subseteq \mathbf{R}^d$. Then the cones of any triangulation of K have an index bounded by*

$$D = \left(\max_{i=1}^{n} \|\mathbf{b}_i\|^2\right)^{n/2}. \tag{19}$$

Proof. Let $B \in \mathbf{Z}^{d\times d}$ be the generator matrix of a full-dimensional cone of a triangulation of K; then the columns B form a subset $\{\mathbf{b}_{i_1}, \dots, \mathbf{b}_{i_d}\} \subseteq \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$. Therefore

$$|\det B| \leq \prod_{k=1}^{d} \|\mathbf{b}_{i_k}\| \leq \left(\max_{i=1}^{n} \|\mathbf{b}_i\|^2\right)^{n/2},$$

giving the desired bound. □

With these preparations, the following corollary is immediate.

Corollary 17. *Let $\mathbf{v} + B\mathbf{R}^n_+$ be a full-dimensional polyhedral cone with apex at $\mathbf{v} \in \mathbf{Q}^d$, whose basis is given by the columns of the matrix $B \in \mathbf{Z}^{d\times d}$. Let D be defined by (19). Let $\hat{\mathbf{v}} \in \mathbf{Q}^d$ and $\rho \in \mathbf{Q}_+$ be the data from Lemma 15 describing the stability cube of $\mathbf{v} + B\mathbf{R}^n_+$. Using these data, construct $\tilde{\mathbf{v}}$ as in Theorem 11. Then the assertions of Theorem 11 hold.*

Algorithm 18 (All-primal irrational Barvinok algorithm)**.**

Input: A polyhedron $P \subset \mathbf{R}^d$ given by rational inequalities; the maximum index ℓ.

Output: The rational generating function for $P \cap \mathbf{Z}^d$ in the form (16).

1. Compute all vertices $\mathbf{v}_i$ and corresponding supporting cones C_i of P.
2. Irrationalize all cones by computing new apex vectors $\tilde{\mathbf{v}}_i \in \mathbf{Q}^d$ from $\mathbf{v}_i$ by Corollary 17.
3. Triangulate $\tilde{\mathbf{v}}_i + C_i$ into simplicial cones $\tilde{\mathbf{v}}_i + C_{ij}$, discarding lower-dimensional cones.
4. Apply Barvinok's signed decomposition to the cones $\tilde{\mathbf{v}}_i + C_{ij}$, until all cones have index at most ℓ.
5. Enumerate the integer points in the fundamental parallelepipeds of all resulting cones to obtain the sets A_i.
6. Write down the formula (16).

7. Computational experiments

Algorithms 12 and 18 have been implemented in a new version of the software package LattE, derived from the official LattE release 1.2 (De Loera et al., 2005a). The new version, called LattE macchiato, is freely available on the Internet (Köppe, 2006). In this section, we discuss some implementation details and show the results of first computational experiments.

7.1. **Two substitution methods.** When the generating function g_P has been computed, the number of lattice points can be obtained by evaluating $g_P(\mathbf{1})$. However, $\mathbf{1}$ is a pole of every summand of the expression

$$g_P(\mathbf{z}) = \sum_{i\in I} \epsilon_i \frac{\sum_{\mathbf{a}\in A_i} \mathbf{z}^{\mathbf{a}}}{\prod_{j=1}^{d}(1 - \mathbf{z}^{\mathbf{b}_{ij}})}.$$

The method implemented in LattE 1.2 (see De Loera et al., 2004b) is to use the *polynomial substitution*

$$\mathbf{z} = ((1+s)^{\lambda_1}, \dots, (1+s)^{\lambda_d}).$$

for a suitable vector $\boldsymbol{\lambda}$. Then the constant coefficient of the Laurent expansion of every summand about $s = 0$ is computed using polynomial division. The sum of all the constant coefficients finally gives the number of lattice points.

Another method from the literature (see, for instance, Barvinok and Pommersheim, 1999) is to use the *exponential substitution*

$$\mathbf{z} = (\exp\{\tau\lambda_1\}, \dots, \exp\{\tau\lambda_d\})$$

for a suitable vector $\boldsymbol{\lambda}$. By letting $\tau \to 0$, one then obtains the formula

$$|P \cap \mathbf{Z}^d| = \sum_{i\in I} \epsilon_i \sum_{k=0}^{d} \frac{\operatorname{td}_{d-k}(\langle \boldsymbol{\lambda}, \mathbf{b}_{i1}\rangle, \dots, \langle \boldsymbol{\lambda}, \mathbf{b}_{id}\rangle)}{k! \cdot \langle \boldsymbol{\lambda}, \mathbf{b}_{i1}\rangle \cdots \langle \boldsymbol{\lambda}, \mathbf{b}_{id}\rangle} \sum_{\mathbf{a}\in A_i} \langle \boldsymbol{\lambda}, \mathbf{a}\rangle^k, \tag{20}$$

where td_{d-k} is the so-called Todd polynomial. In LattE macchiato, the exponential substitution method has been implemented in addition to the existing polynomial substitution; see De Loera and Köppe (2006) for implementation details.

7.2. **Implementation details.** We enumerate the lattice points in the fundamental parallelepiped by computing the Smith normal form of the generator matrix B; see Lemma 5.2 of Barvinok (1993).[1] For computing Smith normal forms, we use the library LiDIA, version 2.2.0. For solving the linear program in Lemma 15, we use the implementation of the revised dual simplex method in exact rational arithmetic in `cddlib`, version 0.94a (Fukuda, 2005). All other computations are done using the libraries NTL, version 5.4 (Shoup, 2005) and GMP, version 4.1.4 for providing exact integer and rational arithmetic.

7.3. **Evaluation of variants of the algorithms.** We compare the variants of the algorithms using test instances that can also be solved without the proposed irrationalization techniques. We consider the test instances `hickerson-12`, `hickerson-13`, and `hickerson-14`, related to the manuscript by Hickerson (1991). They describe simplices in $\mathbf{R}^6$ and $\mathbf{R}^7$ that contain 38, 14, and 32 integer points, respectively. The examples are good test cases for our algorithms because the vertices and cones are trivially computed, and all computation time is spent in the Barvinok decomposition. We show the results in Table 1, Table 2, and Table 3. The tables show results for the following methods:

1. Methods without irrationalization, using polarization to avoid computing with lower-dimensional cones:
   (a) LattE 1.2 (De Loera et al., 2005a), decomposing down to unimodular cones in the dual space (Algorithm 5 with $\ell = 1$).
   (b) Likewise, but using the implementation in the library `barvinok` by Verdoolaege (2006), version 0.21.

[1] The author wishes to thank Susan Margulies for prototyping the enumeration code.

Table 1. Results for `hickerson-12`

	Without irrationalization					With irrationalization		
		Time (s)					Time (s)	
				LattE macchiato			LattE macchiato	
Max. index	Cones	LattE v 1.2	`barv.` v 0.21	Poly	Exp	Cones	Poly	Exp
1	11625	17.9	11.9	10.0	16.7	7929	7.8	12.7
10	4251			6.9	7.0	803	1.9	1.6
100	980			6.9	2.1	84	**1.3**	0.3
200	550			9.1	1.5	76	1.3	0.3
300	474			9.9	1.4	58	1.4	0.3
500	410			11.7	1.3	42	1.6	0.3
1000	130			7.2	0.7	22	1.7	**0.2**
2000	7			**2.2**	0.2	22	1.8	0.2
5000	7			2.8	**0.2**	7	2.8	0.2

(c) LattE macchiato, decomposing cones in the dual space, until all cones in the primal space have at most index ℓ (Algorithm 5), then using polynomial substitution. We show the results for different values of ℓ.
(d) Likewise, but using exponential substitution.

2. Methods with irrationalization, performing triangulation in the dual space and Barvinok decomposition in the primal space (Algorithm 12):
(a) LattE macchiato with polynomial substitution.
(b) LattE macchiato with exponential substitution.

The table shows computation times in CPU seconds on a PC with a Pentium M processor with 1.4 GHz. It also shows the total number of simplicial cones created in the decomposition, using the different variants of LattE; note that we did not measure the number of simplicial cones that the library `barvinok` produced.

We can make the following observations:

i. By stopping Barvinok decomposition before the cones are unimodular, it is possible to significantly reduce the number of simplicial cones. This effect is much stronger with irrational decomposition in the primal space than with decomposition in the dual space.
ii. The newly implemented exponential substitution has a computational overhead compared to the polynomial substitution that was implemented in LattE 1.2.
iii. However, when we compute with simplicial, non-unimodular cones, the exponential substitution becomes much more efficient than the polynomial substitution. Hence the break-even point between enumeration and decomposition is reached at a larger cone index.
The reason is that the inner loops are shorter for the exponential substitution; essentially, only a sum of powers of scalar products needs to be evaluated in the formula (20). This can be done very efficiently.
iv. The best results are obtained with the irrational primal decomposition down to an index of about 500 to 1000 and exponential substitution.

Table 2. Results for `hickerson-13`

Max. index	Without irrationalization: Cones	Time (s): LattE v 1.2	Time (s): `barv.` v 0.21	LattE macchiato: Poly	LattE macchiato: Exp	With irrationalization: Cones	LattE macchiato: Poly	LattE macchiato: Exp
1	466 540	793	589	421	707	483 507	479	770
10	272 922			**345**	428	55 643	117	109
100	142 905			489	249	9 158	**83**	22
200	122 647			625	222	6 150	93	17
300	98 654			903	199	4 674	105	14
500	90 888			1056	193	3 381	137	13
1000	73 970			1648	**190**	2 490	174	**13**
2000	66 954			2166	201	1 857	237	14
5000	49 168			5040	286	1 488	354	18
10000	43 511			7278	370	1 011	772	34

Table 3. Results for `hickerson-14`

Max. index	Without irrationalization: Cones	Time (s): LattE v 1.2	Time (s): `barv.` v 0.21	LattE macchiato: Poly	LattE macchiato: Exp	With irrationalization: Cones	LattE macchiato: Poly	LattE macchiato: Exp
1	1 682 743	4 017	15 284	2 053	3 466	552 065	792	1 244
10	1 027 619			**1736**	2 177	49 632	168	143
100	455 474			2 294	1 089	8 470	**128**	29
200	406 491			2 791	990	5 554	157	22
300	328 340			4 131	875	4 332	187	19
500	303 566			4 911	842	3 464	235	**18**
1000	236 626			8 229	**807**	2 384	337	18
2000	195 368			12 122	817	1 792	481	21
5000	157 496			22 972	1 034	1 276	723	27
10000	128 372			31 585	1 270	956	1 095	38

7.4. **Results for challenge problems.** In Table 4 we show the results for some larger test cases related to Hickerson (1991). We compare LattE 1.2 with our implementation of irrational primal decomposition (Algorithm 12) with maximum index 500. The computation times are given in CPU seconds. The computations with LattE 1.2 were done on a PC Pentium M, 1.4 GHz; the computations with LattE macchiato were done on a slightly slower machine, a Sun Fire V890 with UltraSPARC-IV processors, 1.2 GHz.

Both the traditional Barvinok algorithm (Algorithm 5 with $\ell = 1$) and the homogenized variant of Barvinok's algorithm (De Loera et al., 2004a) do not work well for cross polytopes. The reason is that triangulation is

TABLE 4. Results for larger Hickerson problems

n	d	Lattice points	LattE v 1.2 Cones	LattE v 1.2 Time	LattE macchiato Cones	LattE macchiato Time
15	7	20	293 000	10 min 55 s	2 000	22 s
16	8	54	3 922 000	3 h 35 min	19 000	3 min 56 s
17	8	18			2 655 000	7 h 59 min
18	9	44	61 500 000	77 h 00 min	200 000	49 min 12 s
20	10	74			2 742 000	13 h 05 min

TABLE 5. Results for cross polytopes

d	Without irrationalization Cones	Without irrationalization Time (s)	All-primal irrational Cones	All-primal irrational Time (s)
4	384	1.1		0.9
5	3 840	6.5		1.4
6	46 264	91.7		2.7
7	653 824	1688.7		5.5
8			1 000	12.3
9			2 000	29.6
10			5 000	74.8
11			11 000	189.1
12			24 000	483.0
13			53 000	1 231.2
14			114 000	3 145.6
15			245 000	8 180.9

done in the dual space, so hypercubes need to be triangulated. We show the performance of the traditional Barvinok algorithm in Table 5. We also show computational results for the all-primal irrational algorithm (Algorithm 18 with $\ell = 500$), using exponential substitution. The computation times are given in CPU seconds on a Sun Fire V440 with UltraSPARC-IIIi processors, 1.6 GHz.

A challenge problem related to the paper by Beck and Hoşten (2006), case $m = 42$, could be solved using the all-primal irrational decomposition algorithm (Algorithm 18) with exponential substitution. The method decomposed the polyhedron to a total of 1.1 million simplicial cones of index at most 500. The computation took 66 000 CPU seconds on a Sun Fire V440 with UltraSPARC-IIIi processors, 1.6 GHz. The problem could not be solved previously because the traditional algorithms first tried to triangulate the polar cones, which does not finish within 17 days of computation.

Conclusions and future work

The above computational results with our preliminary implementation have shown that the proposed irrationalization techniques can speed up the Barvinok algorithm by large factors.

A further speed-up can be expected from a refined implementation. For example, the choice of the irrational shifting vector is based on worst-case

estimates. It may be worthwhile to implement a randomized choice of the shifting vector (within the stability cube), using shorter rational numbers than those constructed in the paper. The randomized choice, of course, would not give the same guarantees as our deterministic construction. However, it is easy and efficient to check, during the decomposition, if the generated cones are all irrational; when they are not, one could choose a new random shifting vector (or resort to the one constructed in this paper) and restart the computation.

References

Alexander I. Barvinok. Polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed. *Mathematics of Operations Research*, 19:769–779, 1994.

Alexander I. Barvinok. Computing the volume, counting integral points, and exponential sums. *Discrete Comput. Geom.*, 10(2):123–141, 1993.

Alexander I. Barvinok and James E. Pommersheim. An algorithmic theory of lattice points in polyhedra. In *New Perspectives in Algebraic Combinatorics*, volume 38 of *Math. Sci. Res. Inst. Publ.*, pages 91–147. Cambridge Univ. Press, Cambridge, 1999.

Matthias Beck and Serkan Hoşten. Cyclotomic polytopes and growth series of cyclotomic lattices. eprint arXiv:math.CO/0508136, 2006.

Matthias Beck and Frank Sottile. Irrational proofs for three theorems of Stanley. eprint arXiv:math.CO/0501359 v4, 2005.

Matthias Beck, Christian Haase, and Frank Sottile. Theorems of Brion, Lawrence, and Varchenko on rational generating functions for cones. eprint arXiv:math.CO/0506466, 2005.

Michel Brion. Points entiers dans les polyédres convexes. *Ann. Sci. École Norm. Sup.*, 21:653–663, 1988.

Benno Büeler, Andreas Enge, and Komei Fukuda. Exact volume computation for polytopes: A practical study. In Gil Kalai and Günter M. Ziegler, editors, *Polytopes – Combinatorics and Computation*, volume 29 of *DMV-Seminars*, Basel, 2000. Birkhäuser Verlag.

Jesús A. De Loera and Matthias Köppe. New LattE flavors. Manuscript, 2006.

Jesús A. De Loera, David Haws, Raymond Hemmecke, Peter Huggins, Bernd Sturmfels, and Ruriko Yoshida. Short rational functions for toric algebra and applications. *Journal of Symbolic Computation*, 38(2):959–973, 2004a.

Jesús A. De Loera, Raymond Hemmecke, Jeremiah Tauzer, and Ruriko Yoshida. Effective lattice point counting in rational convex polytopes. *Journal of Symbolic Computation*, 38(4):1273–1302, 2004b.

Jesús A. De Loera, David Haws, Raymond Hemmecke, Peter Huggins, Jeremiah Tauzer, and Ruriko Yoshida. LattE, version 1.2. Available from URL http://www.math.ucdavis.edu/~latte/, 2005a.

Jesús A. De Loera, David Haws, Raymond Hemmecke, Peter Huggins, and Ruriko Yoshida. A computational study of integer programming algorithms based on Barvinok's rational functions. *Discrete Optimization*, 2: 135–144, 2005b.

Jesús A. De Loera, Raymond Hemmecke, Matthias Köppe, and Robert Weismantel. Integer polynomial optimization in fixed dimension. *Mathematics of Operations Research*, 31(1):147–153, 2006a.

Jesús A. De Loera, Raymond Hemmecke, Matthias Köppe, and Robert Weismantel. FPTAS for mixed-integer polynomial optimization with a fixed number of variables. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms, Miami, FL, January 22–24, 2006*, pages 743–748, 2006b.

Komei Fukuda. `cddlib`, version 094a. Available from URL http://www.cs.mcgill.ca/~fukuda/soft/cdd_home/cdd.html, 2005.

GMP, version 4.1.4. The GNU multiple precision arithmetic library. Available from URL http://www.swox.com/gmp/, 2004.

Dean Hickerson. Relatives of the triple and quintuple product identities. Manuscript, 1991.

Ravi Kannan and Achim Bachem. Polynomial algorithms for computing the smith and hermite normal forms of an integer matrix. *SIAM J. Comput.*, 8(4):499–507, November 1979.

Matthias Köppe. LattE macchiato, version 1.2-mk-0.6. Available from URL http://www.math.uni-magdeburg.de/~mkoeppe/latte/, 2006.

LiDIA, version 2.2.0. A C++ library for computational number theory. Available from URL http://www.informatik.tu-darmstadt.de/TI/LiDIA/, 2006.

Victor Shoup. NTL, a library for doing number theory. Available from URL http://www.shoup.net/ntl/, 2005.

Sven Verdoolaege. `barvinok`, version 0.21. Available from URL http://freshmeat.net/projects/barvinok/, 2006.

Sven Verdoolaege, Rachid Seghir, Kristof Beyls, Vincent Loechner, and Maurice Bruynooghe. Counting integer points in parametric polytopes using Barvinok's rational functions, 2005. To appear in Algorithmica.

Otto-von-Guericke-Universität Magdeburg, Department of Mathematics, Institute for Mathematical Optimization (IMO), Universitätsplatz 2, 39106 Magdeburg, Germany

E-mail address: mkoeppe@imo.math.uni-magdeburg.de